DELAY LOCKED LOOP FOR USE IN SEMICONDUCTOR MEMORY DEVICE

Field of the Invention

The present invention relates to a semiconductor memory device; particularly to a time delay locked loop having a short locking time.

Background of the Invention

In general, a delay locked loop represents a circuit that is used to synchronize an internal clock of a synchronous memory device with an external clock without errors. That is, since a timing delay occurs when the external clock is inputted into the inside of the memory device, the delay locked loop is used to control the timing delay to thereby synchronize the internal clock with the external clock.

Referring to Fig. 1, there is illustrated a block diagram of a conventional delay locked loop.

If a clock signal Clock_1 is coupled thereto, a controllable delay chain block 100 delays the clock signal Clock_1 by a predetermined delay time to thereby produce a delayed clock signal Delayed_clock.

The delayed clock signal Delayed_clock is fed to a pre-delay/post-delay comparison block 110 that compares the delayed clock signal Delayed_clock with a reference clock signal Reference_clock to thereby determine whether increasing

or decreasing the predetermined delay time. Through the comparison process, the pre-delay/post-delay comparison block 110 generates output signals Add_delay and Subtract_delay that are, in turn, fed back to the controllable delay chain block 100 so as to adjust the delay time.

The above delay and comparison processes are repeated until a delayed time of the delayed clock signal Delayed_clock is identical to that of the reference clock signal Reference_clock.

Referring to Fig. 2, there is shown a block diagram of a conventional time delay locked loop employing a careful delay controller.

Once a clock signal Clock_1 is coupled thereto, a controllable delay chain block 200 generates a delayed clock signal Delayed_clock by delaying the clock signal Clock_1 and the delayed clock signal Delayed_clock is provided to a pre-delay/post-delay comparison block 210.

The pre-delay/post-delay comparison block 210 compares the delayed clock signal Delayed_clock with a reference clock signal Reference_clock so as to determine whether increasing or decreasing the delay time of the delayed clock signal Delayed_clock. As a result of the comparison process, the pre-delay/post-delay comparison block 210 produces output signals Add_delay and Sub_delay to a careful delay controller 220.

The careful delay controller 220 is employed to preclude an incorrect determination for the delay time, wherein the

incorrect determination may occur by a noise introduced by the power supply or an irregular noise at a system. That is, the delay controller 220 controls the controllable delay chain block 200 to change the delay time only when the delay time determination satisfies a predetermined standard by collecting the results of more than two continuous determination processes instead of directly applying the result of the pre-delay/post-delay comparison block 210 to change the delay time. The output signals Add_delay and Subtract_delay of the delay controller 220 are fed back to the controllable delay chain block 20 to thereby adjust the delay time.

As described above, the conventional delay locked loop is strong to a noise at the state when the delayed locked loop normally operates and the locking is done. However, there is a disadvantage that it takes a very long time from an initial condition in which the locking is not caused to the locking. That is, since, in order to adjust the time delay, there needs at least two determination processes for the increase or decrease in the delay time generated from the pre-delay/post-delay comparison block 210, there is a problem making a time required for the locking much longer compared with that of using only one time of determination.

Summary of the Invention

It is, therefore, a primary object of the present invention to provide a time delay locked loop which classifies

a locking state of the delay locked loop into a locked state and a unlocked state, and either uses a conventional time delay adjusting method in the unlocked state or reduces or eliminates a incorrect determination of the locking introduced by a noise in the locked state.

In accordance with the present invention, there is provided a delay locked loop for use in a semiconductor memory device, comprising: a controllable delay chain block for controlling a delay time of a clock signal coupled thereto; a comparison block for detecting the increase and decrease in the delay time by comparing a reference clock signal with a delayed clock signal generated from the controllable delay chain block; and an instant locking delay control block for controlling the increase and decrease in the delay time of the delay chain block in response to an output signal of the comparison block, the delayed clock signal and the reference clock signal.

Brief Description of the Drawings

The above and other objects and features of the present invention will become apparent from the following description of preferred embodiments given in conjunction with the accompanying drawings, in which:

Fig. 1 illustrates a block diagram of a conventional delay locked loop;

Fig. 2 shows a conventional delay locked loop employing a

careful delay controller;

Fig. 3 is a block diagram of a delay locked loop in accordance with the present invention;

Fig. 4 provides a block diagram of an instant locking delay controller in accordance with the present invention;

Fig. 5 depicts a block diagram of a locking detector in accordance with the present invention;

Fig. 6 represents a block diagram of a pre-delay/post-delay determination block in Fig. 5;

Fig. 7 describes a block diagram of a pre-delay/post-delay logic block in Fig. 5;

Fig. 8 is a block diagram of a locking detector further including an output block in accordance with the present invention;

Fig. 9 shows a block diagram of the output block in accordance with one embodiment of the present invention; and

Fig. 10 is illustrated a block diagram of the output block in accordance with another embodiment of the present invention.

Detailed Description of the Preferred Embodiments

Hereinafter, with reference to the drawings, some of the preferred embodiments of the present invention will be explained in detail.

Referring to Fig. 3, there is illustrated a block diagram of a delay locked loop in accordance with the present

invention.

The delay locked loop comprises a controllable delay chain block 300 for adjusting a delay time of a clock in response to output signals Add_delay and Subtract_delay of an instant locking delay controller 320, a pre-delay/post-delay comparison block 310 for determining the increase and decrease of the delay time by comparing a delayed clock signal Delayed_clock outputted from the controllable delay chain block 300 with a reference clock signal Reference_clock, and the instant locking delay controller 320 for generating the output signals Add_delay and Subtract_delay by using output signals Add_delay_i and Sub_delay_i of the pre-delay/post-delay comparison block 310, the delayed clock signal Delayed_clock and the reference clock signal Reference_clock, wherein the output signals Add_delay and Subtract_delay are used to control the increase and decrease of the delay time at the controllable delay chain block 300.

Unlikely to the conventional delay locked loop described in Fig. 2, in the inventive delay locked loop shown in Fig. 3, the delayed clock signal Delayed_clock and the reference clock signal Reference_clock are coupled to both the pre-delay/post-delay comparison block 310 and the instant locking delay controller 320 in parallel. Therefore, the instant locking delay controller 320 can check whether the locking is accomplished or not by comparing the delayed clock signal Delayed_clock and the reference clock signal Reference_clock.

Referring to Fig. 4, there is provided a block diagram of

the instant locking delay controller 320 in accordance with the present invention.

The instant locking delay controller 320 comprises a conventional careful delay controller 410, a shift multiplexer 420 and a locking detector 430.

The locking detector 430 compares the reference clock signal Reference_clock with the time delayed clock signal Delayed_clock to thereby check whether a time difference between the reference clock signal Reference_clock and the time delayed clock signal Delayed_clock is locked in a predetermined range. The output signals Add_delay_i and Sub_delay_i of the pre-delay/post-delay comparison block are inputted to the careful delay controller 410 that generates output signals Add_delay_int and Sub_delay_int compensating the locking due to a noise. The signals Add_delay_i and Sub_delay_i are also coupled to the shift multiplexer 420.

Then, the shift multiplexer 420 selects one of the inputted signals Add_delay_i and Sub_delay_i and chooses one of the inputted signals Add_delay_int and Sub_delay_int. At this time, if the locking detector 430 determines that the reference clock signal Reference_clock and the time delayed clock signal Delayed_clock are sufficiently locked, the shift multiplexer 420 determines an instruction provided from the careful delay controller 410 as a final time delay instruction, and, if otherwise, the shift multiplexer 420 decides an instruction coupled from the pre-delay/post-delay comparison block 310 as the final time delay instruction.

The locking detector 430 generates a signal Rough_lock_flag controlling the shift multiplexer 420 based on the reference clock signal Reference_clock and the time delayed clock signal Delayed_clock.

Referring to Fig. 5, there is shown an exemplary block diagram of the locking detector 430 in Fig. 4.

The locking detector 430 comprises a first pre-delay/post-delay determination block 530 which receives the reference clock signal Reference_clock and a signal provided from a first delay block 510 which delays the time delayed clock signal Delayed_clock by a predetermined delay time; a second pre-delay/post-delay determination block 540 which receives the time delayed clock signal Delayed_clock and a signal supplied from a second delay block 520 which delays the reference clock signal Reference_clock by a preset delay time; and a pre-delay/post-delay logic block 550 generates the control signal Rough_lock_flag by using output signals Lead_lag_flag_1 and Lead_lag_flag_2 of the first and the second pre-delay/post-delay determination blocks 530 and 540.

The first pre-delay/post-delay determination block 530 is provided with the reference clock signal Reference_clock and the signal S1 outputted from the first delay block 510 delaying the time delayed clock signal Delayed_clock by the predetermined delay time, and compares the reference clock signal Reference_clock and the signal S1. If the reference clock signal Reference_clock is slower than the signal S1, it means that the time delayed clock signal Delayed_clock goes

ahead a predetermined time compared with the reference clock signal Reference_clock. Therefore, in order to determine that the time delayed signal Delayed_clock and the reference clock signal Reference_clock are locked, it should be determined that the signal S1 is slower than the reference clock signal Reference_clock.

The second pre-delay/post-delay determination block 540 is provided with the time delayed clock signal Delayed_clock and the signal R1 outputted from the second delay block 520 delaying the reference clock signal Reference_clock by the preset delay time and compares the time delayed clock signal Delayed_clock and the signal R1. If the signal R1 is faster than the time delayed clock signal Delayed_clock, it means that the time delayed clock signal Delayed_clock is behind the predetermined time compared with the reference clock signal Reference_clock. Accordingly, in order to determine that the time delayed signal Delayed_clock and the reference clock signal Reference_clock are locked, it should be determined that the signal R1 is slower than the time delayed clock signal Delayed_clock.

The pre-delay/post-delay logic block 550 checks whether or not the second pre-delay/post-delay determination block 540 determines that the signal R1 is slower than the time delayed clock signal Delayed_clock and whether or not the first pre-delay/post-delay determination block 530 determines that the signal S1 is slower than the reference clock signal Reference_clock. If all of the above requirements are

satisfied, the pre-delay/post-delay logic block 550 determines that the locking is achieved and, if otherwise, decides that the locking is not achieved. As a result, the selection signal Rough_lock_flag representing whether the locking is achieved or not is outputted to the shift multiplexer 420.

Referring to Fig. 6, there is described an exemplary circuit diagram of the pre-delay/post-delay determination block 530 or 540 in Fig. 5 which is capable of detecting the first and last of falling edges of the reference clock signal Reference_clock and the time delayed clock signal Delayed_clock.

A reference signal REFERENCE is inverted at a first inverter INV1 and the inverted signal is inputted into a first input node of a first NAND gate N1. An output signal of the first NAND gate N1 is fed to a first input node of a second NAND gate N2.

Meanwhile, a clock signal IN_CLOCK is inverted at a second inverter INV2 and then the inverted signal is coupled to a second input node of the second NAND gate N2. An output signal of the second NAND gate N2 is fed to a second input node of the first NAND gate N1.

The inverted signal generated from the first inverter INV1 is provided into a third inverter INV3 and an output signal of the third inverter INV3 is fed to a pulse generator B1. Then, an output signal of the pulse generator B1 is coupled to a first input node of a third NAND gate N3 and a second input node of a fourth NAND gate N4 in parallel.

The output signal of the first NAND gate N1 is provided into a second input node of the third NAND gate N3 and the output signal of the second NAND gate N2 is supplied into a first input node of the fourth NAND gate N4. Output signals of the third and the fourth NAND gates N3 and N4 are provided to a fifth and a sixth NAND gates N5 and N6, respectively. An output signal of the fifth NAND gate N5 is inputted to the sixth NAND gate N6, and an output signal of the sixth NAND gate N6 is fed to the fifth NAND gate N5 to be latched and also generated as a determination signal Lead0_lag1_flag.

The first latch consisting of the first NAND gate N1 and the second NAND gate N2 is a circuit for detecting the first and last of falling edges of the reference signal REFERENCE and the clock signal IN_CLOCK.

The third latch made of the fifth NAND gate N5 and the sixth NAND gate N6 has a function of storing the data of the first latch.

The second latch composed of the third NAND gate N3 and the fourth NAND gate N4 is a gate for delivering the data of the first latch to the third latch.

The pulse generator B1 produces a signal that is enabled to transfer the data of the first latch when the first latch finishes its comparison process.

In Fig. 7, there is depicted a detailed circuit diagram of the pre-delay/post-delay logic block 550 illustrated in Fig. 5.

The determination signal Lead_lag_flag_1 outputted from

the first pre-delay/post-delay determination block 530 is inverted by an inverter INV4 and then coupled to one input node of a NAND gate N7. The signal Lead_lag_flag_2 generated from the second pre-delay/post-delay determination block 540 is fed to the other input node of the NAND gate N7. An output signal of the NAND gate N7 is inverted by an inverter INV5 and then outputted as the selection signal Rough_lock_flag controlling the shift multiplexer 420.

For instance, in the first and the second determination blocks 530 and 540, it is assumed that, if a signal that is compared with a reference signal is faster than the reference signal, an output thereof has a logic low state and, if otherwise, the output has a logic high state. According to the assumption, when the second pre-delay/post-delay determination block 540 determines that the signal R1 is slower than the time delayed clock signal Delayed_clock and the first pre-delay/post-delay determination block 530 decides that the signal S1 is slower than the reference clock signal Reference_clock, the first and the second determination blocks 530 and 540 generate output signals having the logic high state. If otherwise, the first and the second determination blocks 530 and 540 generate output signals having the logic low state.

In Fig. 8, there is shown a block diagram further including an output block 800 in addition to the locking detector 430.

The locking detector 430 is a circuit for checking

whether the locking is accomplished or not and the locking range can be relatively wide. In this case, although the locking detector 430 determines that the locking is achieved, it takes much longer time to accomplish a maximum locking which can be induced by the delay locking loop.

If the careful delay controller 410 is used as soon as the locking detector 430 determines that the locking is accomplished, the entire locking time may become longer. Therefore, information representing that the locking is achieved should be outputted after a given time. On the other hand, information representing that the locking is not accomplished should be outputted without any delay.

In order to utilize the careful delay controller 410 from the time of outputting a signal depicting that the locking is completed after guaranteeing a much smaller locking error by proceeding a locking procedure without directly using the careful delay controller 410 although a signal representing that the locking is accomplished is generated from the pre-delay/post-delay logic block 550, the output block 800 receives the time delayed clock signal Delayed_clock and delays the output signal of the pre-delay/post-delay logic block 550 by one or more than one clock cycles. Further, if information showing that the locking is not accomplished at the pre-delay/post-delay logic block 550 is coupled thereto, the output block 800 immediately deactivates the selection signal Rough_lock_flag of controlling the shift multiplexer to thereby make the careful delay controller 410 not be used.

Referring to Fig. 9, there is described one embodiment of the output block 800. In accordance with this embodiment, several shift registers are connected to each other in series. The output block 800 comprises a first shift register 900 receiving as inputs a reference clock signal CLOCK of synchronizing the registers, the output signal DATA of the pre-delay/post-delay logic block 550 and a reset signal RESET; a plurality of shift registers 910 serially connected to the first shift register 900 and receiving the reference clock signal CLOCK and the reset signal RESET; a NAND gate 920 performing a negative AND operation for the output signal DATA of the pre-delay/post-delay logic block 550 and shifted signals generated from the plurality of shift registers 910; and an inverter 930 outputting the selection signal Rough_lock_flag by inverting an output of the NAND gate 920.

The first shift register 900 receives as its input the signal DATA generated from the pre-delay/post-delay logic block 810 and the inputted signal DATA is shifted to a next shift register in response to the clock signal CLOCK. The shift registers are initialized by the reset signal RESET. The NAND gate 920 generates information representing that the locking is accomplished when the inputted signal DATA passes a predetermined number of shift registers. Information describing that the locking is not accomplished is outputted without passing the shift registers.

In Fig. 10, there is illustrated a block diagram of the output block 800 in accordance with another embodiment of the

present invention.

The output block 800 comprises a multiplicity of shift registers 950 connected to each other in series which receive a clock signal CLOCK of synchronizing registers and the output signal DATA of the pre-delay/post-delay logic block 550 as a reset signal. An input of the first shift register is fixed as a logic high state, and a shifted signal produced from the final shift register is delayed by a delay unit 960 and outputted as the selection signal Rough_Lock_flag.

As described above, in a delay locked loop of a semiconductor memory device in accordance with the present invention, by using difference circuits in cases the locking is completed or not, i.e., using the careful delay controller When the locking is accomplished and not using the careful delay controller when the locking is not completed, there are effects capable of obtaining a short locking time when the locking is not accomplished and reducing an incorrect locking determination due to a noise caused by the careful delay controller when the locking is done.

While the present invention has been described with respect to the particular embodiments, it will be apparent to those skilled in the art that various changes and modifications may be made without departing from the spirit and scope of the invention as defined in the following claims.